IN THE CLAIMS

Please amend claims 18, 26 and 37, as set forth below.

The text of all pending claims, along with their current status, is set forth below:

1-17. (Cancelled)

18. (Currently Amended) A method for managing a queue having a plurality of queue headers within a computer system comprising the steps of:

attaching a plurality of data structures to the plurality of queue headers, where each

data structure is attached to at least one of the plurality of queue headers; and

controlling operations of the plurality of queue headers utilizing ~~one of~~ a <u>function</u>

<u>library containing a</u> plurality of queue function calls, wherein the function calls

are configured to <u>manage the plurality of queue headers operating on the data</u>

<u>structures.</u> ~~search the data structure of the computer system.~~

19-25. (Canceled)

26. (Currently Amended) A computer system that employs a queuing system, the queuing system comprising:

a plurality of generic queue headers, the plurality of generic queue headers being

connected by a plurality of links; and

a data structure attached to at least one of the plurality of generic queue headers

without reference to the plurality of links, wherein the plurality of queue

headers are controlled by <u>a function library containing</u> a plurality of function

calls configured to <u>manage the plurality of queue headers operating on the data</u> <u>structures.</u> ~~search the data structure of the computer system.~~

27. (Original) The computer system set forth in claim 26, comprising a plurality of queue function calls for controlling operations of the plurality of generic queue headers.

28. (Original) The computer system set forth in claim 27, wherein the plurality of queue function calls includes an insert call, a remove call, a search and remove call, a search and insert call, a search only call and a peek call.

29. (Original) The computer system set forth in claim 27, wherein each data structure includes a search key field, and one of the generic queue function calls utilizes a search command to scan each data structure attached to one of the generic queue headers until the search command matches the search key field and the operation of the one of the queue function calls is performed.

30. (Original) The computer system set forth in claim 26, wherein each generic queue header includes a pointer to a next generic queue header, a pointer to a previous generic queue header, and a pointer to the attached data structure.

31. (Original) The computer system set forth in claim 26, wherein each generic queue header comprises a dynamic queue header.

32. (Original) The computer system set forth in claim 26, wherein each generic queue header comprises a static queue header.

3

33. (Original) The computer system set forth in claim 26, wherein each of the plurality of links is uni-directional.

34. (Original) The computer system set forth in claim 26, wherein each of the plurality of links is bi-directional.

35. (Original) The computer system set forth in claim 26, wherein the queuing system comprises a portion of an operating system.

36. (Original) The computer system set forth in claim 26, wherein the queuing system comprises a portion of a driver.

37. (Currently Amended) A method of operating a queuing system, the method comprising the acts of:

linking a plurality of generic queue headers with a plurality of links; and

attaching a data structure to at least one of the plurality of generic queue headers

without reference to the plurality of links, wherein the plurality of queue

headers are controlled by <u>a function library containing</u> a plurality of function

calls configured to <u>manage the plurality of queue headers operating on the data

structures.</u> ~~search the data structure of a computer system.~~

38. (Original) The method set forth in claim 37, comprising the act of controlling operations of the plurality of generic queue headers using a plurality of queue function calls.

39. (Original) The method set forth in claim 38, comprising the act of defining the plurality of queue function calls to include an insert call, a remove call, a search and remove call, a search and insert call, a search only call and a peek call.

40. (Original) The method set forth in claim 38, comprising the acts of:

defining each data structure to include a search key field; and

utilizing a search command to scan each data structure attached to one of the generic queue headers until the search command matches the search key field and the operation of the one of the queue function calls is performed.

41. (Original) The method set forth in claim 37, comprising the act of defining each generic queue header to include a pointer to a next generic queue header, a pointer to a previous generic queue header, and a pointer to the attached data structure.

42. (Original) The method set forth in claim 37, comprising the act of defining each generic queue header to comprise a dynamic queue header.

43. (Original) The method set forth in claim 37, comprising the act of defining each generic queue header to comprise a static queue header.

44. (Original) The method set forth in claim 37, comprising the act of defining each of the plurality of links to be uni-directional.

45. (Original) The method set forth in claim 37, comprising the act of defining each of the plurality of links to be bi-directional.

46. (Original) The method set forth in claim 37, comprising the act of employing the queuing system as a portion of an operating system.

47. (Original) The method set forth in claim 37, comprising the act of employing the queuing system as a portion of a driver.